Project on Liver Disease Prediction

Aim - To Predict the stage of liver Cirrhosis using 18 clinical features. Cirrhosis damages the liver from a variety of causes leading to scarring and liver failure. We will predict the stages of liver Cirrhosis using Machine Learning Models.

Problem Statement - Hepatitis and chronic alcohol abuse are frequent causes of the disease. Liver damage caused by cirrhosis can't be undone, but further damage can be limited. Treatments focus on the underlying cause. In advanced cases, a liver transplant may be required. Predicting the stage of cirrhosis and beginning the treatment before it's too late can prevent the fatal consequences of the disease.

Data-

Train Dataset - It consists of a total of 6801 datapoints.
Test Dataset - You must predict the stage of cirrhosis of 3201 datapoints.

The project is sub-divided following sections. These are:

1.Loading necessary libraries

2.Loading Dataset from a CSV file or from a Table.

3.Summarization of Data to understand Dataset.

4. Visualization of Data to understand Dataset (Plots, Graphs etc.)

5. Data pre-processing and Data transformation.

6. Splitting the data set into independent & dependent sets.

7. Importing the train_test_split model from sklearn. model for splitting data into train & test sets.

8. Importing the SVM model & then training those models with the help of fit ().

9. Predicting the trained models & then checking their accuracy of the model.

10. Then, trained the test dataset with Tain dataset with the help of better accuracy model.

11. Finally, predicted the Stage of Cirrhosis with a new data provided to the Model creating a new data frame using dictionary.

Loading Libraries and Dataset:

```
[1]  #importing the packages
     import numpy as np
     import pandas as pd
     import matplotlib.pyplot as plt
     import seaborn as snr
```

```
[22] train_data=pd.read_csv("/content/train_dataset.csv")
     test_data=pd.read_csv("/content/test_dataset.csv")
```

Data Visualization-

```
# @title Bilirubin vs Cholesterol
data.plot(kind='scatter', x='Bilirubin', y='Cholesterol', s=32, alpha=.8)
plt.gca().spines[['top', 'right',]].set_visible(False)
```

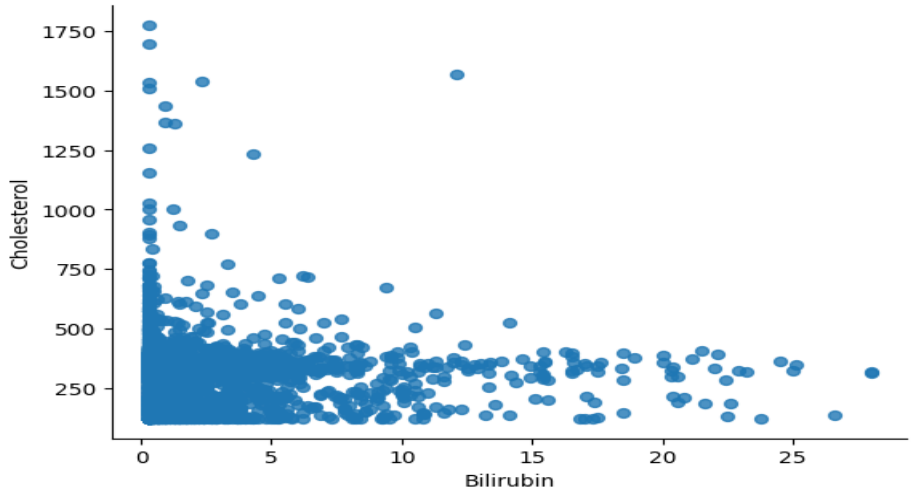

Fig1. Bilirubin vs Cholesterol

Inference: The plot shows that to a certain extent cholesterol increases with bilirubin but remain constant when Bilirubin increases after 10.

```
data.plot(kind='scatter', x='Bilirubin', y='Age', s=32, alpha=.8)
plt.gca().spines[['top', 'right',]].set_visible(False)
```

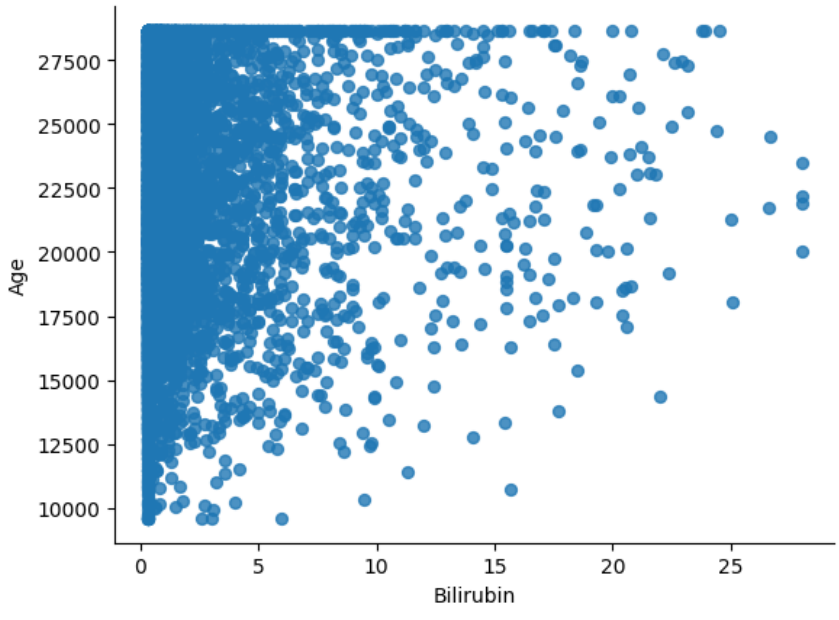

Fig2. Bilirubin vs Age

Inference: Here we are trying to figure out that how Bilirubin reacts with the sample Age, it increases up to 10 with the increase of Age and gradually starts to decrease.

```
[ ]  #checking the number of male and female using a countplot.
     data['Sex'].value_counts().plot.bar(color='peachpuff')
```

```
<Axes: >
```

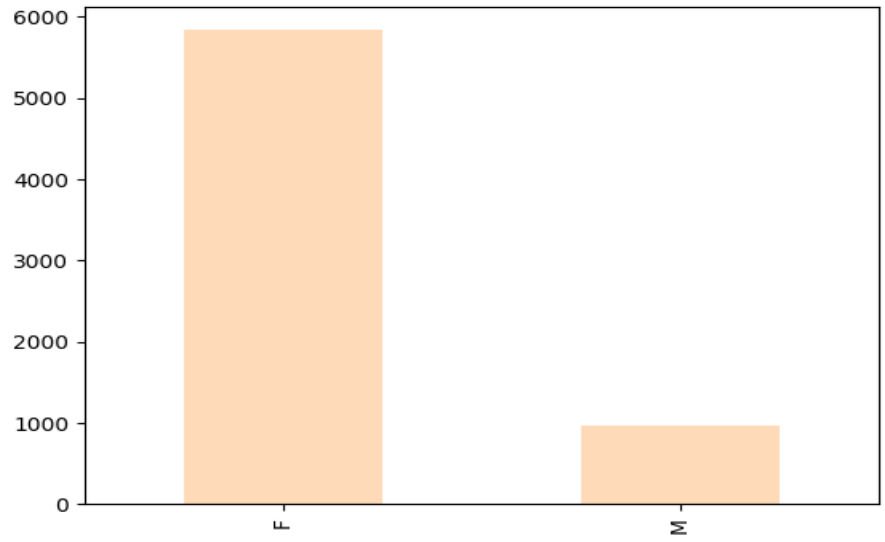

Fig3. Male vs Female count

Inference: Here we are figuring out the number of Males and Females in the dataset provided.

```
df=data[['Sex','Stage']]
df.iloc[0:20].value_counts().plot.bar(color='peachpuff')
```

```
<Axes: xlabel='Sex,Stage'>
```

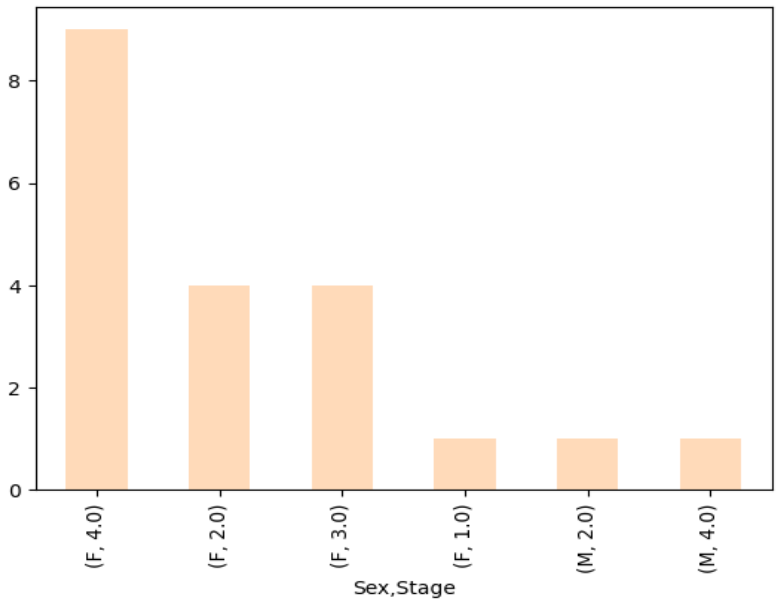

Fig4. Count Plot showing Stage of Cirrhosis based on Gender or sex.

Inference: This plot shows the different stages of Cirrhosis amongst Male and Female patients.

```
f, ax = plt.subplots(figsize=(5,5))
snr.scatterplot(x="Albumin", y="Bilirubin",color='green',data=data);
plt.show()
```

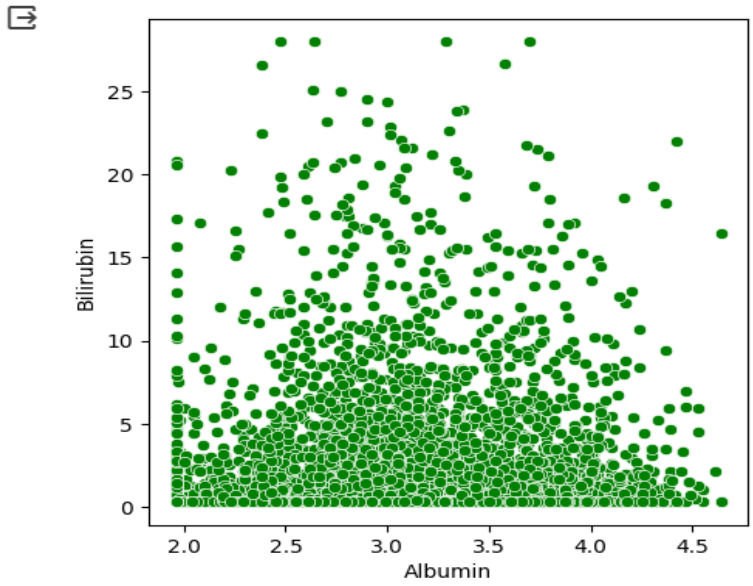

Fig5. Scatter Plot showing Corelation between Bilirubin and Albumin in the samples.

Inference: This plot shows the cluster of Albumin and Bilirubin, where Bilirubin shows an increasing trend with the Albumin levels in the sample.

```
[ ]  plt.figure(figsize=(8,6))
     data.groupby('Sex').sum()['Albumin'].plot.bar(color='midnightblue')
```

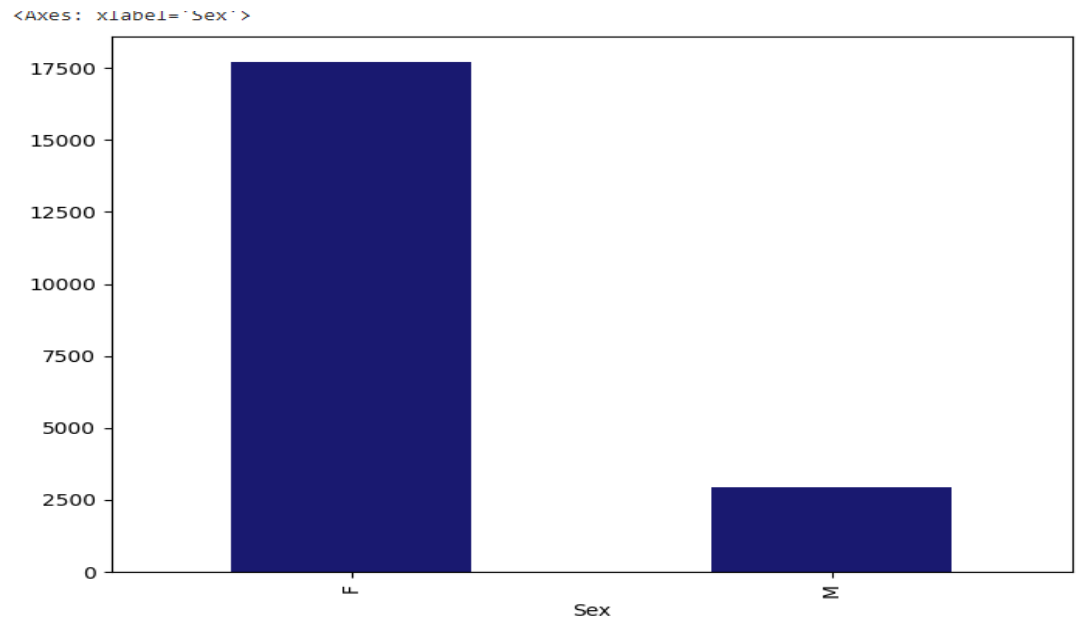

Fig6. Bar graph showing Albumin levels with respect to Sex.

Inference: We can clearly see in the graph that; Albumin is more in Female patients as compared to Male patients.

```
plt.figure(figsize=(10,10))
data.groupby('Prothrombin').sum()['Bilirubin'].plot.bar(color='midnightblue')
```

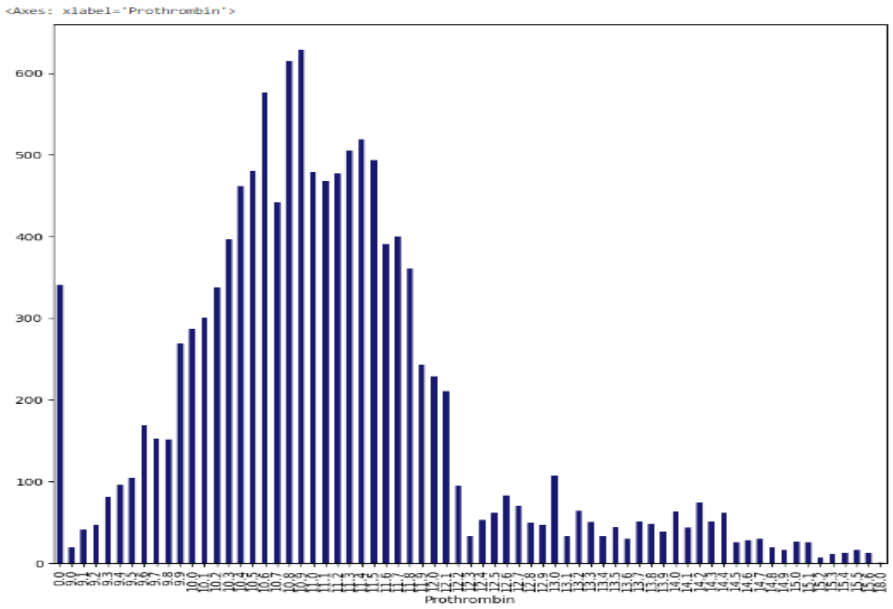

Fig7. Bar graph showing corelation between Prothrombin and Bilirubin levels.

Inference: Here is another interactive graph () that shows Bilirubin increases with the increasing Prothrombin levels and gradually starts to decrease with the increasing levels of Prothrombin.

```
plt.figure(figsize=(10,10))
data.groupby('Prothrombin').sum()['Cholesterol'].plot.bar(color='midnightblue')
```

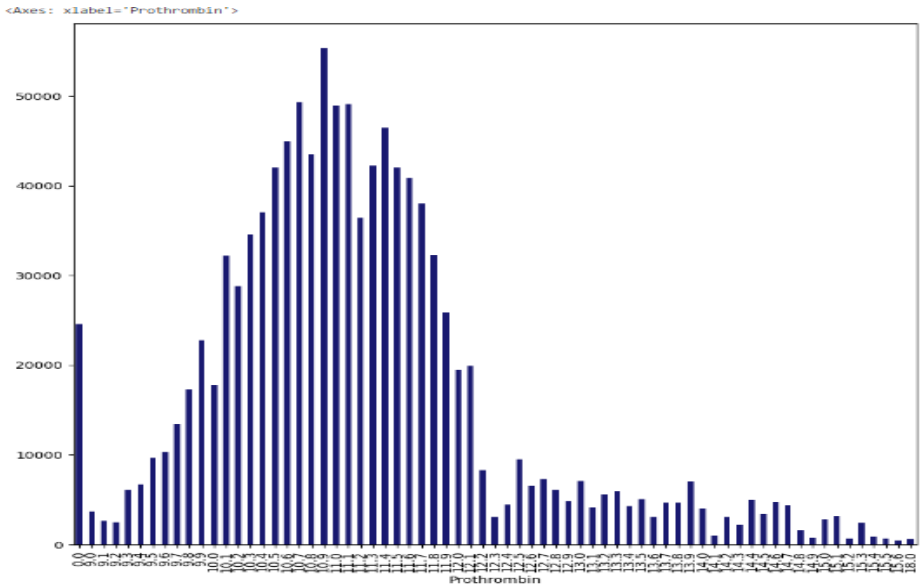

Fig8. Bar graph showing corelation between Prothrombin and Cholesterol levels.

Inference: Here is another interactive graph () that shows Cholesterol increases with the increasing Prothrombin levels and gradually starts to decrease with the increasing levels of Prothrombin.

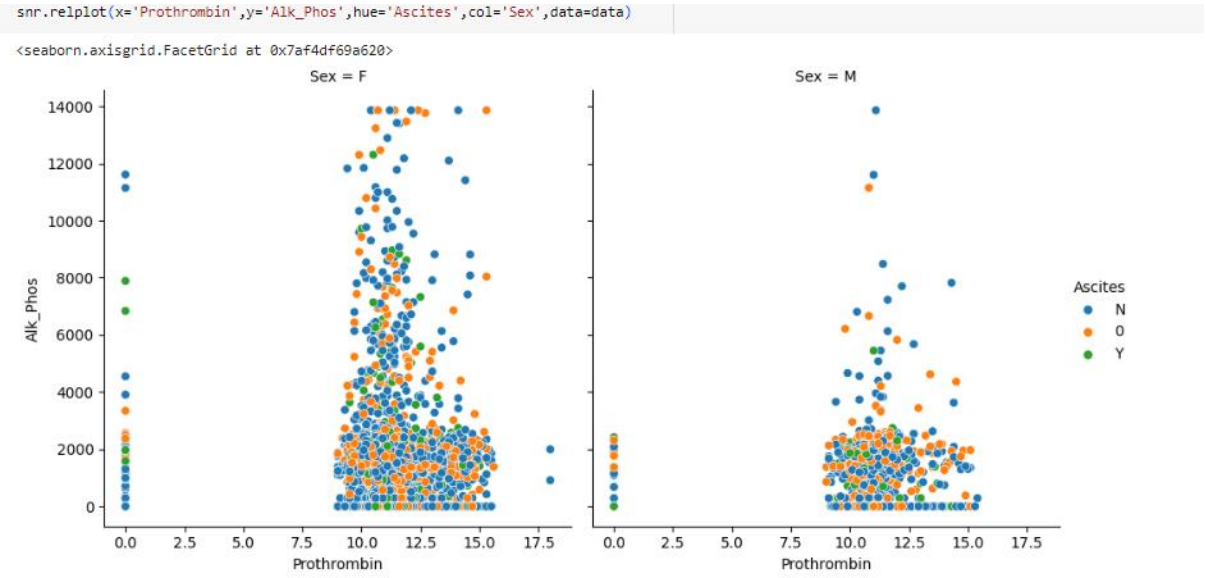

Fig9.Relational plot showing Prothrombin vs Alk Phosphorous vs Ascites.

Inference: In stage 3 of liver cirrhosis, Ascites may occur, it happens due to the increased pressure in portal vein which can cause fluid to accumulate in abdomen. The plot shows Alk-Phos increases after Prothrombin level reaches 9 and keeps on increasing (For Female). The plot also shows a mix of Ascites level in Female and Male both.

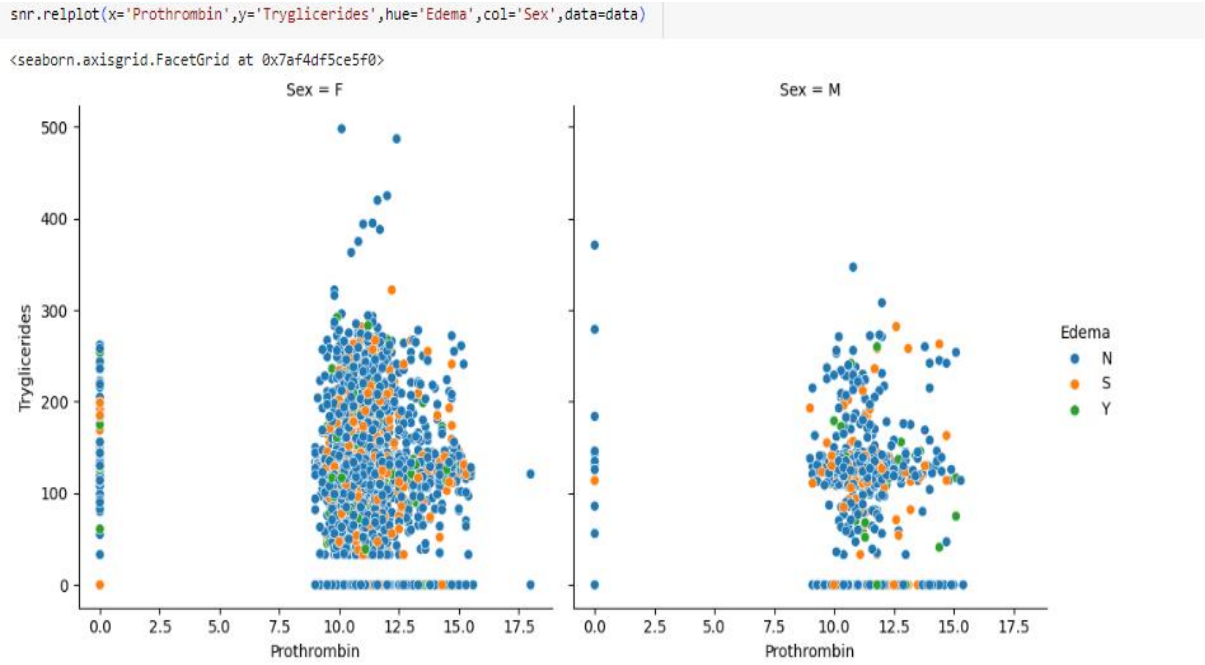

Fig10. Relational plot showing Prothrombin vs Tryglicerides vs Edema.

Inference: In stage 3 of liver cirrhosis, Edema may occur, it happens due to the increased pressure in portal vein which can cause fluid to accumulate in legs. The plot shows Tryglicerides increases after Prothrombin level reaches 9 and keeps on increasing for Female and Male. The plot also shows less levels of Edema in Female and Male both.

```
data.plot.scatter(x='Alk_Phos',y='Age',c='Tryglicerides')
```

```
<Axes: xlabel='Alk_Phos', ylabel='Age'>
```

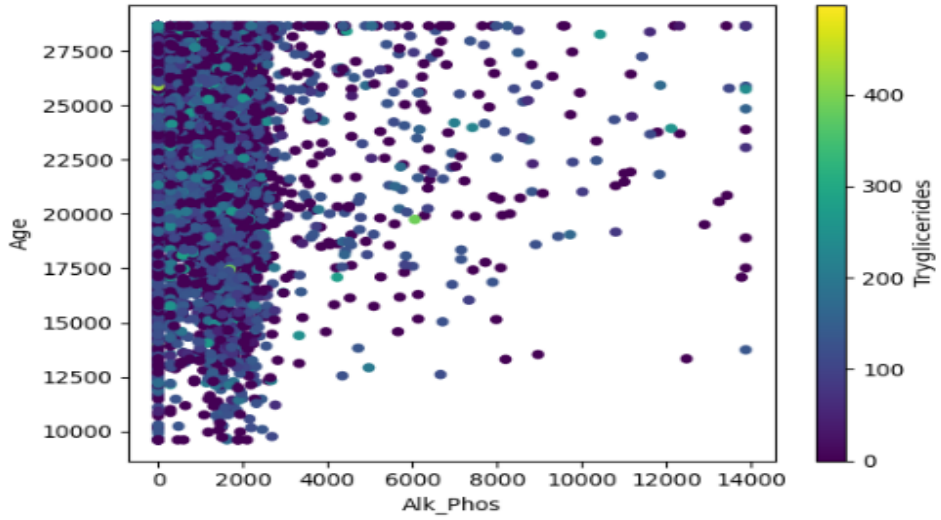

Fig11. Scatter plot showing Alk_Phos vs Age vs Tryglicerides.

Inference: The plot shows as the Age increases the Alk_Phos level increases however after certain point it becomes constant. Tryglicerides level does not show much corelation in this case.

```
[ ] data.plot.scatter(x='Cholesterol',y='Prothrombin',c='Tryglicerides')
```

```
<Axes: xlabel='Cholesterol', ylabel='Prothrombin'>
```

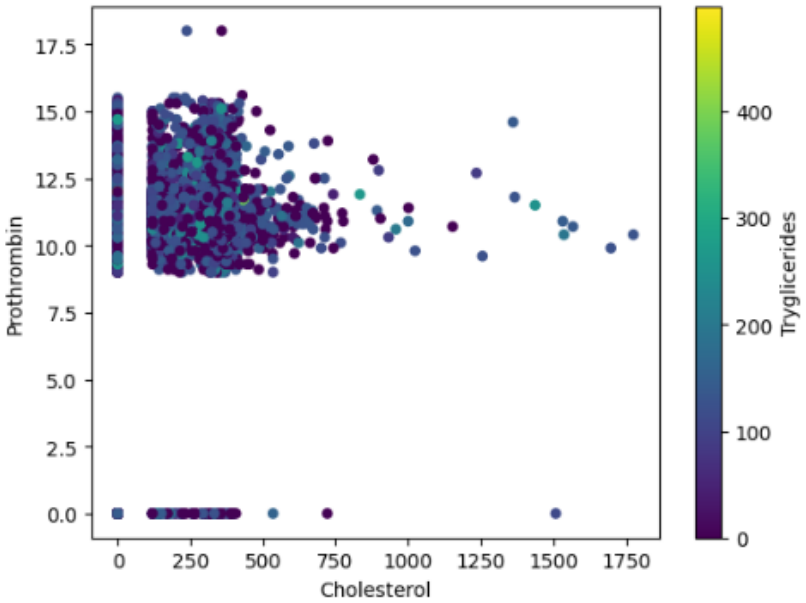

Fig12. Scatter plot showing Prothrombin vs Cholesterol vs Tryglicerides.

Inference: The plot shows very less changes with increasing Prothrombin and Cholesterol however they show a gradual increase in level after Cholesterol level reaches 150.Tryglicerides level does not show much corelation in this case.

Data Cleaning:

One Hot Coding to replace categorical data with numeric values.

```
[ ]  train_data['Status']=train_data['Status'].apply({'C':1,'CL':2,'D':3}.get)
     train_data['Drug']=train_data['Drug'].apply({'D-penicillamine':1,'Placebo':2}.get)
     train_data['Sex']=train_data['Sex'].apply({'M':1,'F':2}.get)
     train_data['Ascites']=train_data['Ascites'].apply({'N':0,'Y':1}.get)
     train_data['Hepatomegaly']=train_data['Hepatomegaly'].apply({'N':0,'Y':1}.get)
     train_data['Spiders']=train_data['Spiders'].apply({'N':0,'Y':1}.get)
     train_data['Edema']=train_data['Edema'].apply({'N':0,'Y':1,'S':2}.get)
```

```
test_data['Status']=test_data['Status'].apply({'C':1,'CL':2,'D':3}.get)
test_data['Drug']=test_data['Drug'].apply({'D-penicillamine':1,'Placebo':2}.get)
test_data['Sex']=test_data['Sex'].apply({'M':1,'F':2}.get)
test_data['Ascites']=test_data['Ascites'].apply({'N':0,'Y':1}.get)
test_data['Hepatomegaly']=test_data['Hepatomegaly'].apply({'N':0,'Y':1}.get)
test_data['Spiders']=test_data['Spiders'].apply({'N':0,'Y':1}.get)
test_data['Edema']=test_data['Edema'].apply({'N':0,'Y':1,'S':2}.get)
```

Checking for the Nan values present in the data.

```
[ ]  check_nan = train_data.isna().values.any()
     print(check_nan)
```

```
True
```

```
[ ]  count_nan = train_data.isna().sum()
     print(count_nan)
```

```
ID                0
N_Days            0
Status            0
Drug           2025
Age               0
Sex               0
Ascites        2246
Hepatomegaly   2427
Spiders        2590
Edema             0
Bilirubin         0
Cholesterol    3101
Albumin           0
Copper         2156
Alk_Phos       2498
SGOT           2102
Tryglicerides  2812
Platelets       338
Prothrombin     155
Stage             0
dtype: int64
```

Replacing the Nan values.

```
[ ]  train_data = train_data.replace([np.nan, -np.inf], 0)
```

```
[ ]  test_data = test_data.replace([np.nan, -np.inf], 0)
```

Splitting Data in to Independent and Dependent Columns.

```
[31] x=train_data[['ID', 'N_Days', 'Status', 'Drug', 'Age', 'Sex', 'Ascites',
            'Hepatomegaly', 'Spiders', 'Edema', 'Bilirubin', 'Cholesterol',
            'Albumin', 'Copper', 'Alk_Phos', 'SGOT', 'Tryglicerides', 'Platelets',
            'Prothrombin']]

     y=train_data[['Stage']]
```

```
[32] x_test=test_data[['ID', 'N_Days', 'Status', 'Drug', 'Age', 'Sex', 'Ascites',
            'Hepatomegaly', 'Spiders', 'Edema', 'Bilirubin', 'Cholesterol',
            'Albumin', 'Copper', 'Alk_Phos', 'SGOT', 'Tryglicerides', 'Platelets',
            'Prothrombin']]
```

Creating Machine Learning Model with Support Vector Machine Classifier.

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,train_size=0.80)
```

```
from sklearn import svm
SVC=svm.SVC(kernel='poly') #rbf, linear, poly, sigmoid
SVC.fit(x_train,y_train)
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/utils/validation.py:1143: DataConversionWarning: A column-vector y was
  y = column_or_1d(y, warn=True)
SVC(kernel='poly')
```

```
predictions=SVC.predict(x_test)
```

```
from sklearn.metrics import accuracy_score,confusion_matrix
cm=confusion_matrix(y_test,predictions)
ac=accuracy_score(y_test,predictions)
```

```
print(cm)
```

```
[[  0   0   0 109]
 [  0   0   0 282]
 [  0   0   0 288]
 [  0   0   0 681]]
```

```
print(ac)
```

```
0.5007352941176471
```

Here we can see that Support Vector Machine (SVM) ML Model is giving an Accuracy of 50%.

Now we will classify a new patient and predict that the patient is in which stage of Liver Cirrhosis.

```
#Classification of a new Patient
my_data={'ID':2484, 'N_Days':1100, 'Status':2, 'Drug':2, 'Age':24640, 'Sex':1, 'Ascites':1,
      'Hepatomegaly':1, 'Spiders':1, 'Edema':1, 'Bilirubin':3, 'Cholesterol':240,
      'Albumin':2.5, 'Copper':50, 'Alk_Phos':1905.7, 'SGOT':110, 'Tryglicerides':150, 'Platelets':195,
      'Prothrombin':11.1}

index=[1]
new_data=pd.DataFrame(my_data,index)
```

```
new_data
```

	ID	N_Days	Status	Drug	Age	Sex	Ascites	Hepatomegaly	Spiders	Edema	Bilirubin	Cholesterol	Albumin	Copper	Alk_Phos	SGOT	Tryglicerides	Platelets	Prothrombin
1	2484	1100	2	2	24640	1	1	1	1	1	3	240	2.5	50	1905.7	110	150	195	11.1

```
#classification of a new employee
classification=SVC.predict(new_data)
print('The new employee will have',classification,'Stage of Cirrhosis')
```

```
The new employee will have [4.] Stage of Cirrhosis
```

Output: The new patient is in Stage 4 of Liver Cirrhosis.

Conclusion: The liver patients data set was used to implement prediction and classification algorithms, which in turn reduces the workload on doctors. We employed machine learning techniques to examine the patient's total liver condition. When a training data set is available, our proposed classification scheme can significantly enhance classification performance. Then, using a machine learning classifier, good and bad values are classified. Here, the output of the proposed classification model showed less accuracy in predicting the result.

However, some of the future directions to improve the accuracy of liver disease prediction and classification models is to include more diverse data sources, improving liver disease prediction and classification is to combine multiple machine learning techniques, machine learning models can be trained to predict the likelihood of liver disease and the stages of Cirrhosis in individuals based on their unique characteristics.